

REMARKS

The Office Action mailed on November 07, 2006 has been received and reviewed. Claims 1-37 remain in the case. Claims 1-37 were rejected under 35 U.S.C. 102(b) as being anticipated by Wu et al. (6,732,267), hereinafter simply Wu.

In light of the rejections, a review of the present invention may help clarify the novelty of the Applicants' claims over the cited prior art. As shown in Figures 3-9 and elsewhere, the present invention updates a code image on a communications adapter. A new code image is loaded into memory on the communications adapter, the memory concurrently storing a copy of the old code image used by the communications adapter along with the new code image. A memory initialization module invokes the new code image to perform a memory initialization operation while the old code image is still executing. The new code image is validated by an error checking procedure or other validity check. The new code image and old code image are then compared to determine an incompatibility between the old code image and new code image.

Incompatibilities between the new code image and old code image may be determined and reconciled. The exact instructions that may be executed to reconcile the incompatibility may depend on the type of incompatibility that needs reconciling. An incompatibility may arise when a difference between the old code image and the new code image causes the new code image to fail to perform at least one function to the performance level of the old code image (see paragraph 63 of Applicants' specification). More specifically, an incompatibility with the new code image may include being a different version than the old code image, using data structures differently, or having different initialization requirements (see paragraphs 63, 64, and 65 of Applicants' specification).

Identifying incompatibilities may include examining the old code image against the new code image to determine if functional differences exist. If the functional difference is reconcilable, then reconciling incompatibilities may adjust the new code image to be compatible. For example, reconciling the incompatibility may comprise changing the order of initialization for storage registers, memory, or hardware devices, or converting the format of a data structure in order for the new code image to use the data structure (see paragraph 51 of Applicants' specification). Another example of reconciling the incompatibility includes "adjust[ing] data structures and configuration settings so that the new code image adequately performs at least the same functions as the old code

image.” (see paragraph 66 of Applicants’ specification). Still other types of operations may be executed to reconcile further types of incompatibilities.

If incompatibilities between the old code image and new code image do not exist or can be resolved before the host adapter is initialized, the host adapter may be initialized using a fastload initialization sequence instead of a standard initialization sequence. The fastload initialization sequence minimizes the amount of time necessary to initialize the host adapter, for example, by assuming that memory and/or hardware are already in a valid state because these states are verified prior to the code overlay. Additionally, to further minimize downtime, the old code image continues to process I/O requests while the new code image is loaded and reconciled. To allow this to happen, the old code image remains in tact during the loading and reconciling of the new code image (see paragraphs 16 and 72 of Applicants’ specification).

Incompatibilities between the old code image and new code image that cannot be resolved before the host adapter is initialized are resolved during the initialization sequence. Invoking the memory initialization code in the new code image facilitates handling the specific initialization sequences required by the new code image. The new code image may require initialization steps unanticipated by a standard initialization sequence. For example, the new code image may include new data structures in a memory location previously unused by the old code image that must be initialized.

Applicants would like to note that determining and reconciling incompatibilities between an old code image and a new code image as described in Applicants’ specification is not synonymous with validating the new code image. Validating the new code image merely includes verifying the new code image’s integrity using procedures such as a “checksum” procedure.

Applicants assert that the cited prior art is not directed to determining incompatibilities between the old code image and the new code image and does not reconcile the incompatibilities between the old code image and the new code image. Specifically, Wu discloses a method for updating the system BIOS of a target system that verifies the validity of the new code image by executing a “checksum” procedure. By executing the “checksum” procedure, Wu merely validates the integrity of the new code image. Wu does not compare the means and functions of the new code image with the old code image, and therefore does not determine incompatibilities between the code

images (see column 4, lines 27 to 32 of Wu). Wu neither identifies nor reconciles incompatibilities that exist in the new code image. As a result, Wu may allow a new code image that passes the “checksum” procedure but contains incompatibilities to be loaded.

Applicants assert that Wu does not disclose “identifying an incompatibility between the old code image and the new code image.” Rather, Wu discloses “examin[ing] the updated BIOS image to determine whether it is valid by executing any suitable ‘checksum’ procedure, and if so, will proceed to replace the old system BIOS” (see column 4, lines 28-30 of Wu). As indicated above, applicants assert that executing a “checksum” procedure merely verifies the integrity of the image. Conversely, determining an incompatibility includes examining and comparing the means and functions of the old code image with the new code image to determine whether the new code image includes incompatibilities with the target system.

Applicants also assert that Wu does not disclose “reconciling the incompatibility between the old code image and the new code image” Wu discloses updating the BIOS image when validity of the new BIOS image is verified. Similarly, Wu discloses not updating the BIOS image when validity of the new BIOS image is not verified. If the BIOS image is not verified, a series of actions are performed to ensure the faulty BIOS image is not used as an update. Applicants assert that determining whether or not to update the BIOS image as disclosed in Wu is not synonymous with reconciling an incompatibility between the old code image and the new code image. Reconciling an incompatibility allows a new code image that would otherwise work improperly or not at all with the target system to work properly. Applicants assert that Wu fails to teach reconciliation of such an incompatibility.

Regarding the rejection of claim 2, Applicants assert that Wu does not disclose “identify[ing] a characteristic of the old code image and determine[ing] a difference between the old code image and the new code image.” As detailed above, Wu merely discloses determining the validity of the new code image by executing a “checksum” procedure. The disclosed “checksum” procedure only validates the new code image; in addition, the BIOS examines the new code image using the “checksum” procedure (see column 4, lines 27-29 of Wu). But the examination of the new code image by the BIOS is restricted to mere validation. In contrast, Applicants’ assert that their claimed invention not only validates the new code image (see paragraph 59 of Applicants’ specification), but

also determines whether the new code image includes incompatibilities (see paragraphs 49 of Applicants' specifications).

To further illustrate the differences between validating the new code image by executing a "checksum" procedure and determining incompatibilities between the new code image and the old code image, several examples of determining incompatibilities follow. First, an incompatibility may be determined by examining both code images and comparing their version numbers to see if they are from different versions (see paragraphs 49 and 63 of Applicants' specifications). Second, an incompatibility may be determined if the size, location, or other characteristic of the old code image is different from that of the new code image (see paragraph 49 of Applicants' specifications). Third, an incompatibility may be determined if the data structures used by the new code image are different from the data structures used by the old code image (see paragraph 64 of Applicants' specifications). And fourth, an incompatibility may be determined if the new code image comprises different initialization requirements than the old code image (see paragraph 65 of Applicants' specifications). The above illustrations show that merely validating the integrity of the new code image using a "checksum" procedure is different than determining whether an incompatibility between the old code image and the new code image exists.

Regarding the rejection of claim 3, Applicants assert that Wu does not disclose "reconcil[ing] an incompatibility between the old code image and the new code image." Rather, the reference from Wu cited by the examiner refers to the use of an indicator in the BIOS update process upon unsuccessful validation of the new code image.

Applicants assert that the process disclosed by Wu is different than reconciling an incompatibility between the old code image and the new code image. Reconciling an incompatibility may include "adjust[ing] data structures and configuration settings so that the new code image adequately performs at least the same functions as the old code image." (see paragraph 66 of Applicants' specifications). Reconciling an incompatibility is directed towards adjusting either the target system or the new code image to obtain functional cohesiveness.

In order to expedite allowance of Applicant's claims, Applicants have elected to amend claim 1 to include the limitations previously included in dependent claims 2 and 3. Applicants have also

amended each of the other independent claims in similar fashion in order to place the claims in condition for allowance.

CONCLUSION

Applicants assert that Wu does not disclose all of the limitations included in independent claims 1, 15, 16, 23, 24, and 37. Applicants therefore assert that claims 1-37 are in condition for allowance and respectfully request prompt allowance of the pending claims. Wu fails to disclose a fastload code image update that minimizes downtime while concurrently processing I/O requests. Applicants assert that their claimed invention minimizes downtime by determining and reconciling incompatibilities in a new code image while keeping the old code image intact in order to continue processing I/O requests. In the event that the Examiner finds any remaining impediments to the prompt allowance of any of these claims which could be clarified in a telephone conference, the Examiner is respectfully urged to initiate the same with the undersigned.

Respectfully submitted,

Date: February 7, 2007

Kunzler & Associates
8 E. Broadway, Suite 600
Salt Lake City, Utah 84111
Telephone: 801/994-4646

/Brian C. Kunzler/

Brian C. Kunzler
Reg. No. 38,527
Attorney for Applicant